



# Memoria Blog Tecnológico

1. Introducción.....	2
2. Estructura HTML.....	2
3. Estilos CSS.....	2
4. Conclusión.....	2
5. Webgrafía.....	2

# 1. Introducción

En este documento se presentará la memoria del ejercicio evaluable 1, el blog tecnológico, dónde se explicará el HTML y CSS de la página junto a capturas de pantalla y cómo se han llegado a esas soluciones. Además se dejarán los enlaces de donde se consiguió la información.

## 2. Estructura HTML

La página siguen las reglas de estructura de los HTML (header, nav, main, aside y footer):

```
<header>
  <h1>Mi Blog de Tecnología</h1>
  <nav>
    <ul>
      <li><a href="#">Inicio</a></li>
      <li><a href="#">Categorías</a></li>
      <li><a href="#">Acerca de</a></li>
      <li><a href="#">Contacto</a></li>
    </ul>
  </nav>
</header>
<main>
```

En el header aparece el título de la página y el menú de navegación dónde llevará a los demás enlaces de la página.

```
<main>
  <section>
    <h2>El futuro de CSS: Flexbox vs Grid</h2>
    <p>Publicado el 19 de Octubre de 2025</p>
    <p>Una inmersión profunda en las herramientas de maquetación que están revolucionando el diseño web...</p>
    <p>Leer más...</p>
  </section>
  <section class="entradas">
    <h3>Últimas entradas</h3>
    <article>
      <h4>Nuevas API de JavaScript</h4>
      <p>Las últimas novedades que debes conocer en ES2025</p>
    </article>
    <article>
      <h4>Diseño Responsive con solo 10 líneas</h4>
      <p>Trucos y consejos para optimizar tu código.</p>
    </article>
  </section>
</main>
```

El main se divide por sections con cada bloque de contenido, además el segundo section contiene artículos para dividir las diferentes secciones y poder introducir un hueco entre ellas.

```
<aside>
  <section>
    <article>
      <h3>Categorías</h3>
      <p>Web (25)</p>
      <p>Movil (12)</p>
      <p>DevOps (8)</p>
      <hr>
    </article>
    <article>
      <h3>Síguenos</h3>
      <p>Encuentra más contenido en nuestras redes sociales.</p>
      <hr>
    </article>
  </section>
</aside>
```

El aside contiene también sections y articles para dividir los diferentes elementos. En el primer article se podría haber usado tanto como una lista desordenada (<ul>) como varios párrafos. Me decidí por la segunda ya que quitaba de forma natural la sangría que tienen las listas.

```
<footer>
  <p>&copy; 2025 Mi Blog Flexible. Todos los derechos reservados</p>
</footer>
```

Por último está el footer el cual cuenta con los derechos de copyright de la página.

### 3. Estilos CSS

```
/* Reset y fuente global */
*
{
  margin: 0;
  font-family: Arial, Helvetica, sans-serif;
}
```

Quitamos los márgenes que vienen por defecto en el navegador y establecemos que la fuente sea Arial.

```
/* Fondo principal de la página */
body
{
    background-color: #d3d3d3;
}
```

Establecemos un fondo grisáceo para la página.

```
/* Header con barra superior con logo y menú, elementos alineados horizontalmente y centrados verticalmente */
header
{
    padding: 30px;
    display: flex;
    justify-content: space-between;
    align-items: center;
    background-color: #2c3e50;
    color: white;
    height: 50px;
}
```

Hacemos que el header sea flex para que el menú de navegación y el título salgan en línea, además, hacemos justify-content para mandar el título a la izquierda y el menú a la derecha.

```
/* Menú horizontal con efecto al tener el cursor en un enlace */
header li
{
    list-style: none;
    display: inline-block;
    padding: 10px;
}
header li a
{
    text-decoration: none;
    color: whitesmoke;
}
header li a:hover
{
    color: rgb(80, 80, 80);
}
```

Creamos un menú de navegación a partir de una lista desordenada y hacemos que esté en línea.

```
/* Main flotante a la izquierda para permitir aside a la derecha */
main
{
    padding: 30px 0 0 150px;
    float: left;
}
```

Hacemos que main sea un float y lo establecemos a la izquierda.

```

/* Bloques de contenido con fondo blanco y bordes redondeados */
section
{
    padding: 40px 30px;
    background-color: white;
    border-radius: 10px;
}
section h2
{
    color: #354A60;
}
section p
{
    padding: 10px 0;
}

```

Editamos el primer bloque para que sea igual al de la imagen.

```

/* Clase entradas para tener un contenedor flexible de artículos horizontales */
.entradas
{
    background-color: gainsboro;
    display: flex;
    gap: 20px;
}
.entradas article
{
    background-color: white;
    padding: 40px 15px;
    border-radius: 5px;
}
.entradas article h4
{
    padding-bottom: 20px;
}

```

Hacemos que el segundo bloque a través de una clase sea display flex y separamos los elementos con gap.

```

/* Aside flotante a la derecha, con secciones internas y altura fija */
aside
{
    float: right;
    margin: 30px 250px 0 0;
}

```

Hacemos que el aside sea también un float y lo establecemos a la derecha.

```
aside section
{
  background-color: #ECF0F1;
  padding: 0 20px;
  height: 425px;
}
aside h3
{
  color: #45A3DF;
  padding: 30px 0;
}
aside p
{
  padding: 0;
}
/* Línea horizontal con separación */
hr
{
  margin: 20px 10px;
}
```

Editamos el aside con los elementos para que quede igual a la imagen. Restablecemos el padding antes editado de los párrafos.

```
/* Footer centrado */
footer
{
  display: flex;
  align-items: center;
  justify-content: center;
  color: white;
  height: 80px;
  background-color: #2C3E50;
  clear: both;
}
```

Por último, centramos el párrafo del footer con un display flex y hacemos un clear: both para que vaya al final de la página.

## 4. Conclusión

He intentado imitar lo máximo posible la página de la imagen pero he tenido problemas con los floats, debido a que al hacer zoom el float pasa a la línea de abajo al ser de diferentes contenedores, aun así pienso que he conseguido el objetivo del ejercicio usando los diferentes display flex, floats y elementos pedidos en este.

## 5. Webgrafía

### Display Flex:

<https://lenguajecss.com/css/flex/alinear-centrar-css/>

(Apuntes de clase)

### Float:

(Apuntes de clase)

<https://developer.mozilla.org/es/docs/Web/CSS/float>